## Uber Trips Analysis in Python

**Description -**

Utilization of Uber trip data may serve technicians as a convenient means of having sharpened technical knowledge, thus leading to enhanced services and improved customer satisfaction. The proposed study is a python-based project that seeks to inform on popular associations and patterns among the variables found in Uber’s ride data.

For the sensing of the data, the trip data of uber has to be first loaded to the python environment. By taking advantage of library frameworks such as Pandas, data pulling and column relabeling are facilitated which allows for a fast forward of data analysis in a Data Frame.

At the next stage, verification is following and we will start looking into the amount of data. Our dilemmas consist of checking the inputs for their purely inventory nature, managing missing values, and interpreting variable types. Descriptive statistics are used here for the clear forwarding of important data which are patterned by the central tendency and variability levels.

Consecutively, we move on to the part of our approach where we develop a tool to investigate the intraclass relationships within the dataset Consequently, it requires finding coefficients of correlation among the numerical variables to distinguish direct/ inversely correlated patterns. Apart from this, we also study the inter-correlation aspect of the categorical variables by including the dependency and association layer.

Relating the data with visual (graphical representation) tools are possible with libraries like Matplotlib or Seaborn. This entails building scatter plots, histograms, and box plots on the various data relationships. Such crosstabs are the significant contributors in mapping of movie performances, distribution their overall trends as well.

During the exploratory process, the research team will be paying close attention to the brander who may be a trendsetter or any abnormality that requires further scrutiny. Such cases render interesting proceeds in the understanding of freak rides or data problems hiding behind them.

Investigating the patterns and trends of Uber trips may reveal some key points behind how Uber can effectively tie up the services and bring the client experience to a new level. With the help of Python-driven analytics, we uncover underpinning causes that are not directly reflected in the data making data-driven decision-making possible.

A key process of the analysis is decoding a corporate riding figure. To do that i would need to monitor the concentration of plasma and, also,identify specific peaks that may occur in hours and locations where there is higher demand. By applying this approach, it is possible, in detail, not only to identify the process of providing the service but also to assume the role of resource allocation and the pricing strategies, thereby optimizing service efficiency.

Moreover, the consideration of this data can lead to an improved level of studies in which specific groups are offered suitable marketing campaigns as well as services that are related to them as a result. This central aspect could enable a multidimensional analysis to reveal hidden patterns, including dataset improvement.

Visualization not only does reveal the kinds of the connections before our eyes, but also does it in a deep way which gives us an understanding of the connections transparently and intuitively. Spatially and temporal symbolic charts like scatter plots, histograms and heat maps accentuate spatial and temporal dynamics and act as tool for informed decision-making.

First of all, let´s focus on analyzing the code segments.

**Now Let’s look at the code snippets -**

To read the data set we will be importing pandas as pd

import pandas as pd

to visualize the dataset we will be importing matplot library

import matplotlib.pyplot as plt

Reading the Dataset

uber\_df= pd.read\_csv("uber-raw-data-sep14.csv")

Now , Displaying the first 5 records of the Dataset from starting and from ending

uber\_df.head(5)

uber\_df.tail()

Finding the shape of the dataset:

the number of rows and the number of columns in the DataFrame uber\_df.

uber\_df.shape

Now retrieving the Information the Dataset Such as Datatype of each column , number of non null values , memory usage etc .

uber\_df.info()

Now changing the data type of the Date/Time in the column from string to Datetime

uber\_df['Date/Time']= pd.to\_datetime(uber\_df['Date/Time'])

this line of code extracts the day component from the "Date/Time" column and assigns it to a new column named "Day"

ber\_df["Day"] = uber\_df["Date/Time"].apply(lambda x: x.day)

This line of code extracts the hour component from the "Date/Time" column and assigns it to a new column named "Hour"

uber\_df["Hour"] = uber\_df["Date/Time"].apply(lambda x: x.hour)

This line of code extracts the weekday component from the "Date/Time" column and assigns it to a new column named "Weekday"

uber\_df["Weekday"] = uber\_df["Date/Time"].apply(lambda x: x.weekday())

Now again displaying first 5 entries of the dataset

uber\_df.head(5)

Generating a histogram to visualize the density of Uber trips per day

fig,ax = plt.subplots(figsize = (12,6))

plt.hist(uber\_df.Day, width= 0.6, bins= 30)

plt.title("Density of trips per Day", fontsize=16)

plt.xlabel("Day", fontsize=14)

plt.ylabel("Density of rides", fontsize=14)

Generating a histogram to visualize the density of Uber trips per weekday as we know the highest rides are during the weekday

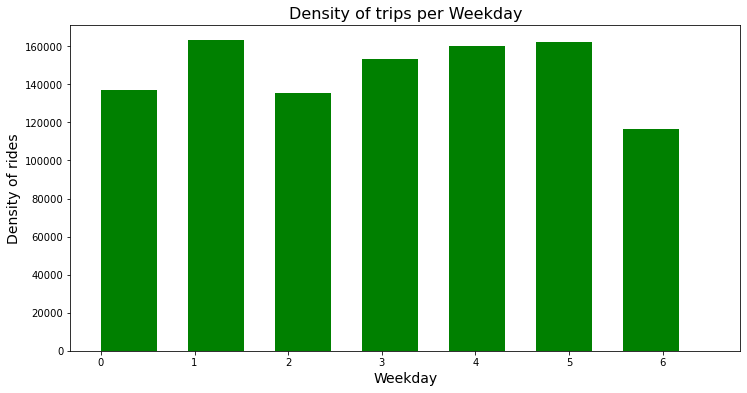
fig,ax = plt.subplots(figsize = (12,6))

plt.hist(uber\_df.Weekday, width= 0.6, range= (0, 6.5), bins=7, color= "green")

plt.title("Density of trips per Weekday", fontsize=16)

plt.xlabel("Weekday", fontsize=14)

plt.ylabel("Density of rides", fontsize=14)



Through the above histogram we got to know that The busiest day in the week for Uber is Monday. On the other hand, Saturday is the day with the least number of rides.

Generating a histogram to visualize the density of Uber trips per hour

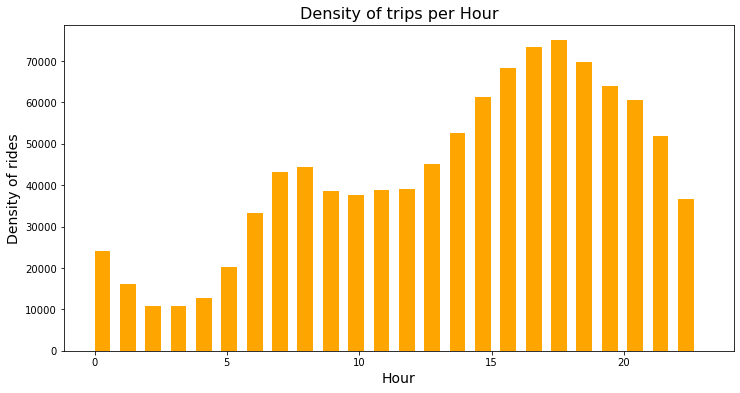
fig,ax = plt.subplots(figsize = (12,6))

plt.hist(uber\_df.Hour, width= 0.6, bins=24, color= "orange")

plt.title("Density of trips per Hour", fontsize=16)

plt.xlabel("Hour", fontsize=14)

plt.ylabel("Density of rides", fontsize=14)



the number of rides decrease gradually from 1 AM to 4 PM and then increases starting from 5 AM onward till it reaches 6 PM which is the hour with the highest number of rides.

Creating a scatter plot to visualize the density of Uber trips based on longitude and latitude coordinates.

fig,ax = plt.subplots(figsize = (12,6))

x= uber\_df.Lon

y= uber\_df.Lat

plt.scatter(x, y, color= "purple")

plt.title("Density of trips per Hour", fontsize=16)

plt.xlabel("Hour", fontsize=14)

plt.ylabel("Density of rides", fontsize=14)



The region with the highest density of rides is near Manhattan and Newburgh. While the region with the lowest density is near New Jersey.